# Ryan Zhang

(236) 863-2246 | ryanzhang@outlook.com | linkedin.com/in/rmzhang | github.com/rzhang57 | ryanz.dev

## Education

**University of British Columbia** — **Vancouver, BC**
*Bachelor of Science in Computer Science* — *May 2028*

- Awards: Trek Excellence Scholarship **(Top 5% of Faculty)**, Charles and Jane Banks Scholar, Science Scholar, Dean's List

## Work Experience

**Mercury** — **New York, NY**
*Software Engineer Intern – Risk Engineering **[Incoming]*** — *Sept 2026 – Dec 2026*

- Building fraud-research agent, **RL** pipelines for fraud detection ML models, with integrated tool calling to core **Haskell** systems

**Microsoft** — **Vancouver, BC**
*Software Engineer Intern – Teams, Low Latency Live Streaming* — *May 2026 – Aug 2026*

- Led development of per-user watermarks across Teams' town-hall live-streaming stack at **0ms** added delay vs re-encoding
- Developed a **C#/.NET** runtime object pub/sub delivery pipeline for consistent event config delivery to **100k** concurrent clients
- Built recording config persistence **REST API** using **C#, CosmosDB (NoSQL)**, syncing policy at publish time to VOD playback API
- Integrated auth into a **TypeScript React** context for email-overlay component, failing safe to audio-only for unverified users
- Implemented middleware service policies driving feature-flagged, ring-based rollouts for town-hall watermark scheduling
- Won the **Microsoft Vancouver intern hackathon** for Cognitive Copilot, a real-time AI meeting accessibility tool

**Planview** — **Vancouver, BC**
*Software Engineer Intern – Copilot, AI Product and Infrastructure* — *Jan 2025 – Aug 2025*

- **Led** agent infrastructure, building **Java/Spring Boot** workflow execution engine running multi-step agentic processes as per-session stateful threads, scaling horizontally on AWS EKS for **100+** enterprise clients; featured at Planview Connect 2025
- Accelerated host-app integrations by **10x** by developing a hybrid **MCP** client supporting both legacy tool calls and MCP servers
- Boosted per-session user retention by **60%** by developing an LLM-driven relevant follow-up question generation pipeline
- Reduced average response times from **10s to 5s** by parallelizing multi-tool invocations with a thread-pooled task executor.
- Integrated GraphQL and AWS SES microservice to enable agentic sending of context-aware emails from user chat history
- Built logging pipeline with data obfuscation and retention compliance using AWS CloudWatch and Fluent Bit log tailing

## Projects

**Noteblock** | *Go, SQLite, PostgreSQL, Electron, React, Ollama* — GitHub

- Built cross-platform desktop note-taking app with modular block system (markdown, code, image annotations, drawing)
- Developed two-layered server architecture - local **Go** process with **SQLite** for sub-100ms user-facing performance; cloud Go service managing user auth, multi-device sync, and conflict resolution enabling seamless offline-online access
- Designed extensible JSON schema for block data storage enabling simpler plugin integrations and open-source contributions
- Integrated on-device AI via **Ollama** API enabling privacy and performance focused AI features keeping info off external servers

**Cognitive Copilot** | *TypeScript, Next.js, React.js, Microsoft Foundry, Azure* — Microsoft Vancouver Hackathon Winner

- Built a real-time meeting assistant to ease cognitive load for ADHD, ESL, and new-hire users; voted overall **winner** by engineers
- Architected the simplified feed, integrating local **Whisper** transcription with a buffer that flushes at semantic pauses into a **two-stage LLM context pipeline** that filters to surface only key points at **sub-2s** delay, at a **10x** increase in relevance
- Developed action-item extraction that builds a to-do list with **Microsoft Graph API** integration for one-click meeting scheduling

## Skills

**Languages**: C++, C#, Java, TypeScript, Go, Python, SQL, JavaScript, HTML, CSS
**Frameworks/Libraries**: Spring Boot, Spring AI (MCP), .NET, React.js, Electron, Node.js, Flask, Next.js, Mockito, Jest
**Databases**: PostgreSQL, SQLite, CosmosDB (NoSQL)
**Developer Tools**: Docker, Git, GitHub Actions, AWS (EKS, EC2, S3, CloudWatch, Bedrock), Kubernetes, ArgoCD, Datadog, Linux